Cortically-Inspired Computing

Neuro-Inspired Computing Elements Workshop, Albuquerque, NM

Mikko H. Lipasti

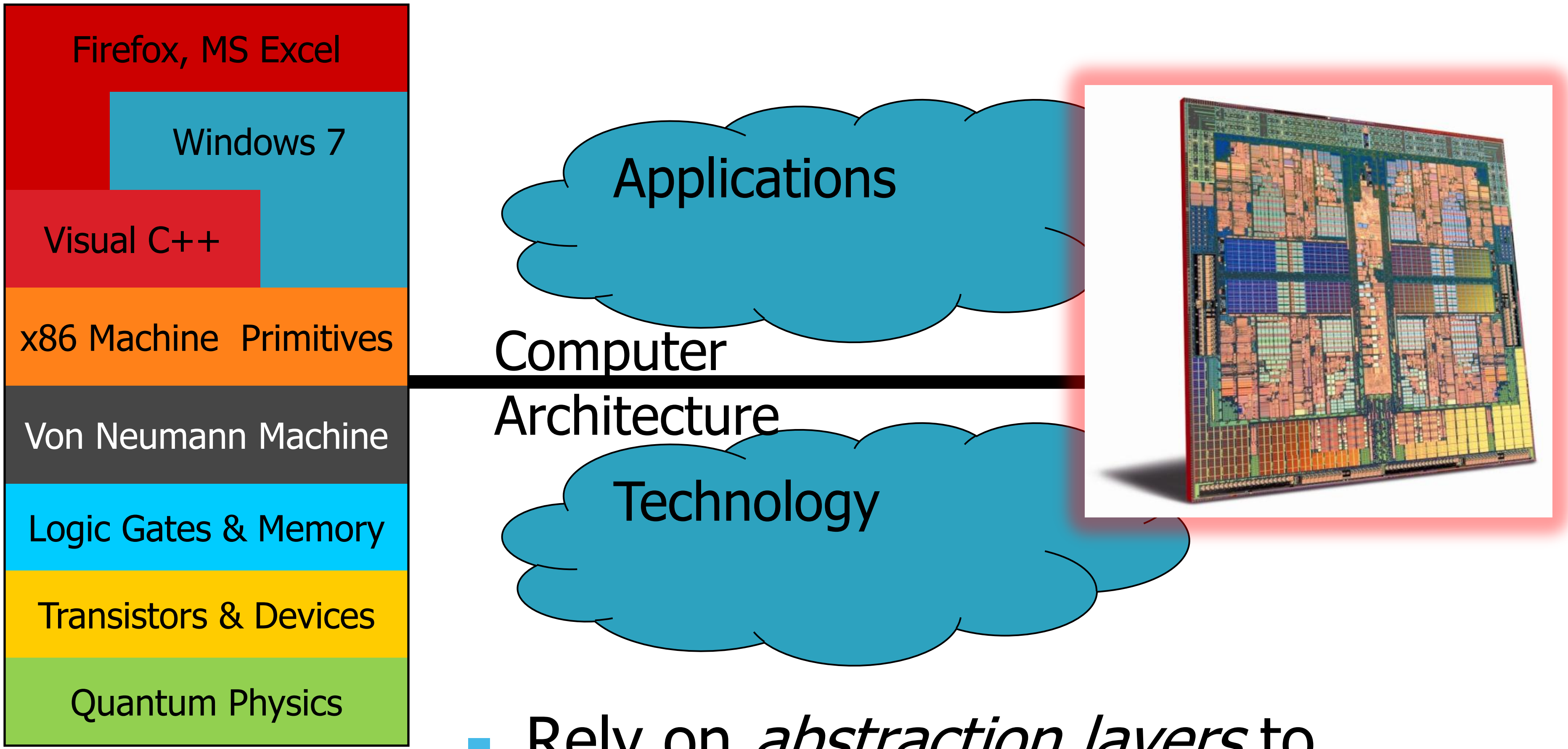
Professor, Electrical and Computer Engineering

University of Wisconsin – Madison

Collaborators: **Atif Hashmi, Andy Nere**, Giulio Tononi , James Thomas (WI); Olivier Temam, Hugues Berry (INRIA); IBM Synapse team; Tianshi Chen, Yunji Chen (ICT); Marc Duranton (CEA); Qi Guo (IBM China); Shi Qiu (USTC); Michele Sebag (LRI);

http://pharm.ece.wisc.edu

What Do I Do?



- Rely on *abstraction layers* to manage complexity
  - *Von Neumann Machine*

End of Moore's Law

- We are running into physical limits
  - Ultimately, single molecule/atom/electron
- Before we reach the atomic scale
  - Manufacturing yield (working parts)
  - Reliability (intermittent/permanent failure)
  - Variability (each device has unique characteristics)
  - Power (can't afford to use all devices all the time)

- On the software side: multicore impact
  - Parallel software is very difficult to write

- Need fundamentally new approaches
  - Von Neumann machines: too successful

Look to Biology?

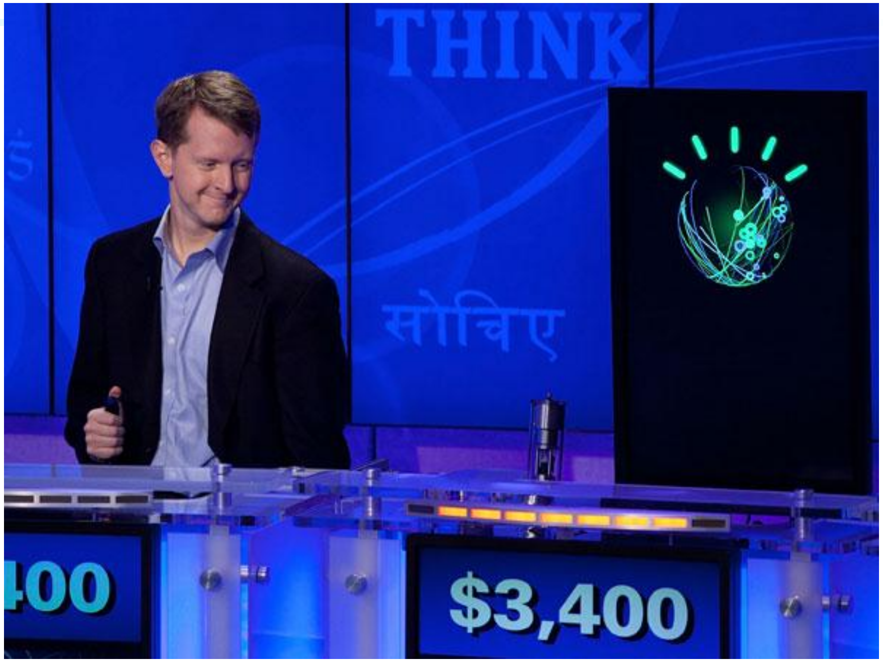
- By no means a novel inspiration

  "If I haven't seen further, it's from standing in the footprints of giants."

- But, neuroscientific understanding has improved substantially
  - Detailed characterization of low-level primitives
  - Structure and connectivity much better understood
  - Advances in measurement, analysis
  - Etc.

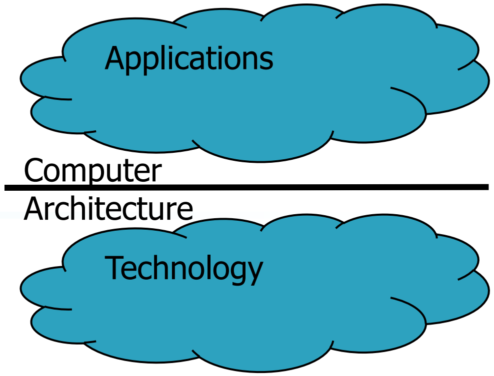
- Is the brain even an interesting candidate?

Ken Jennings vs. IBM Watson



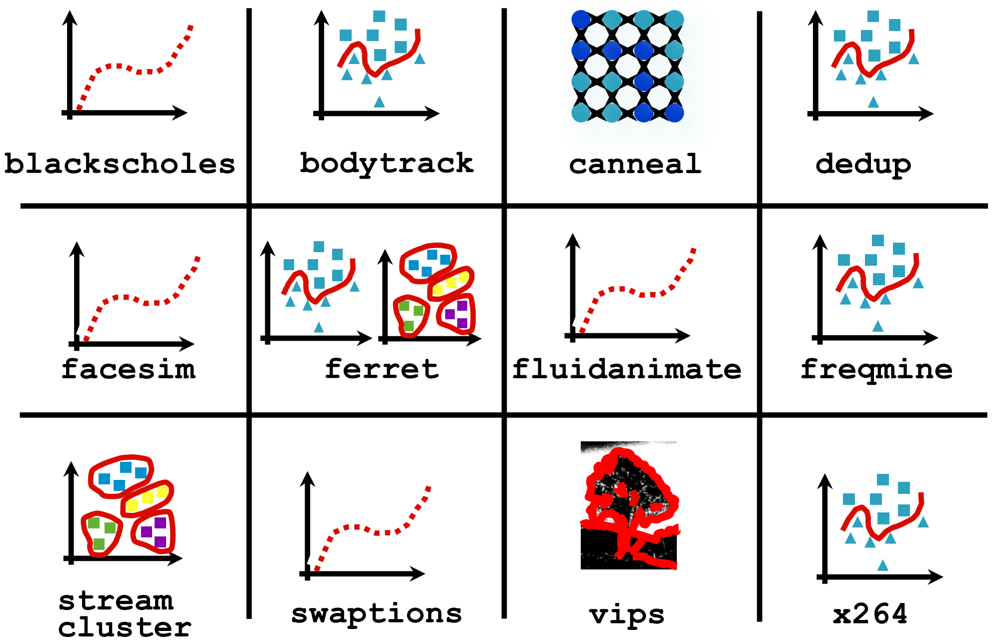
Ken ("baseline")	Watson
Pretty good at Jeopardy (also, life)	Pretty good at Jeopardy
400g gray matter	10 racks, 15TB DRAM, 2880 CPU cores, 80 TFLOPs
30W	200KW
1 lifetime of experience	100 person-years to develop

Talk Outline

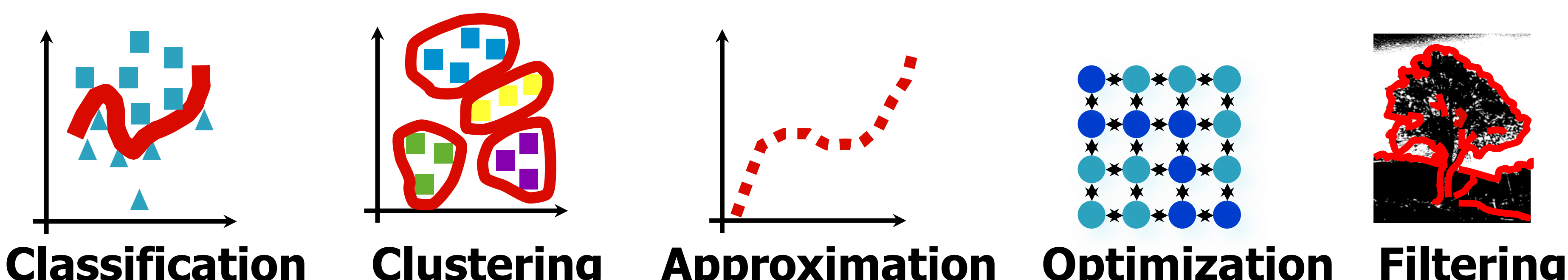


- Introduction & Motivation
- Neuromorphic applications [IISWC'12]
- Semantic Gap in Neuromorphic Systems
  - Neuromorphic ISA proposal [ASPLOS'11]
  - Digital LIF Spiking Neurons [HPCA'13]
- Conclusions & Future Work

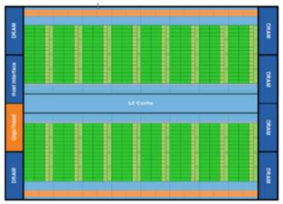
Emerging Applications: RMS



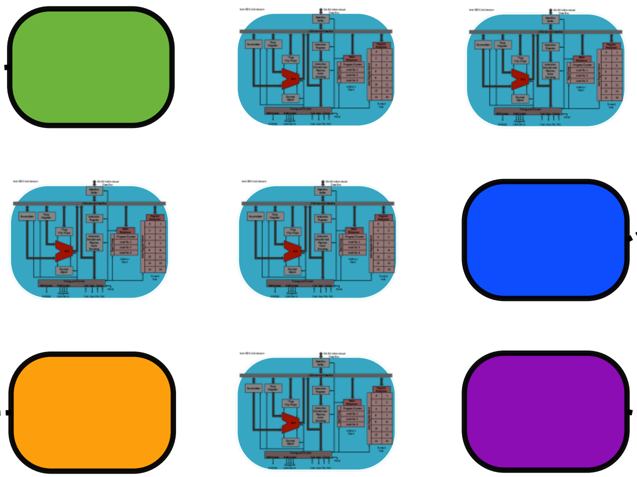
PARSEC [Intel, Princeton]



Application Accelerators

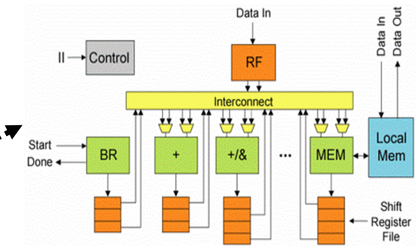


GPUs

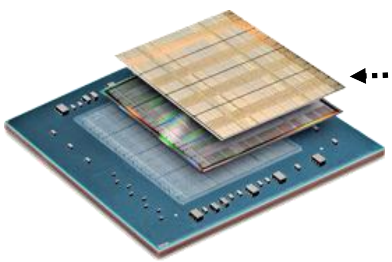


Multi-Purpose Accelerators

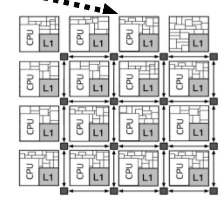
Heterogeneous multi-cores



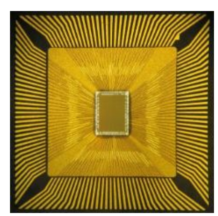
Loop Accelerators



FPGAs/CGRAs



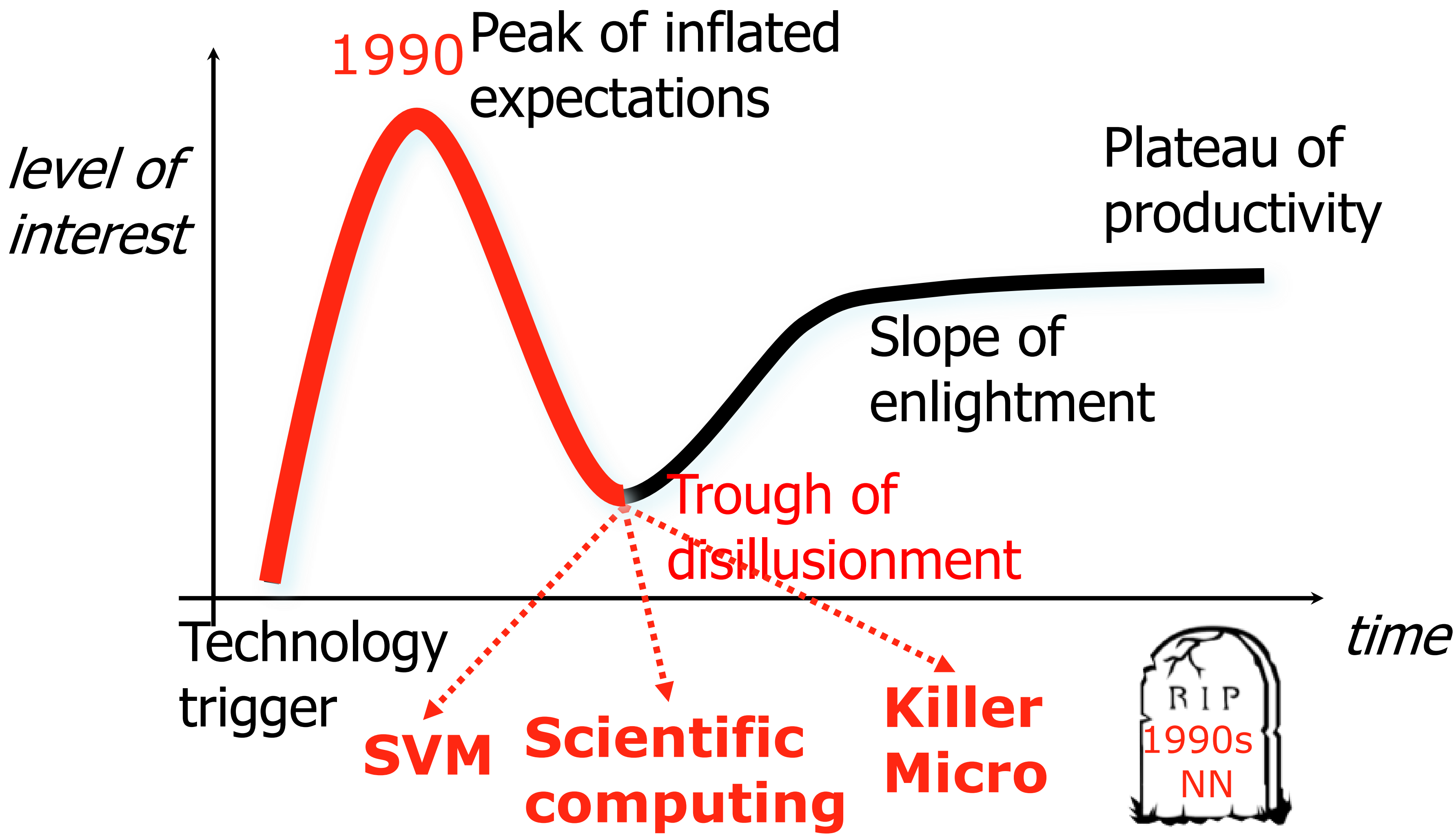
GreenDroid



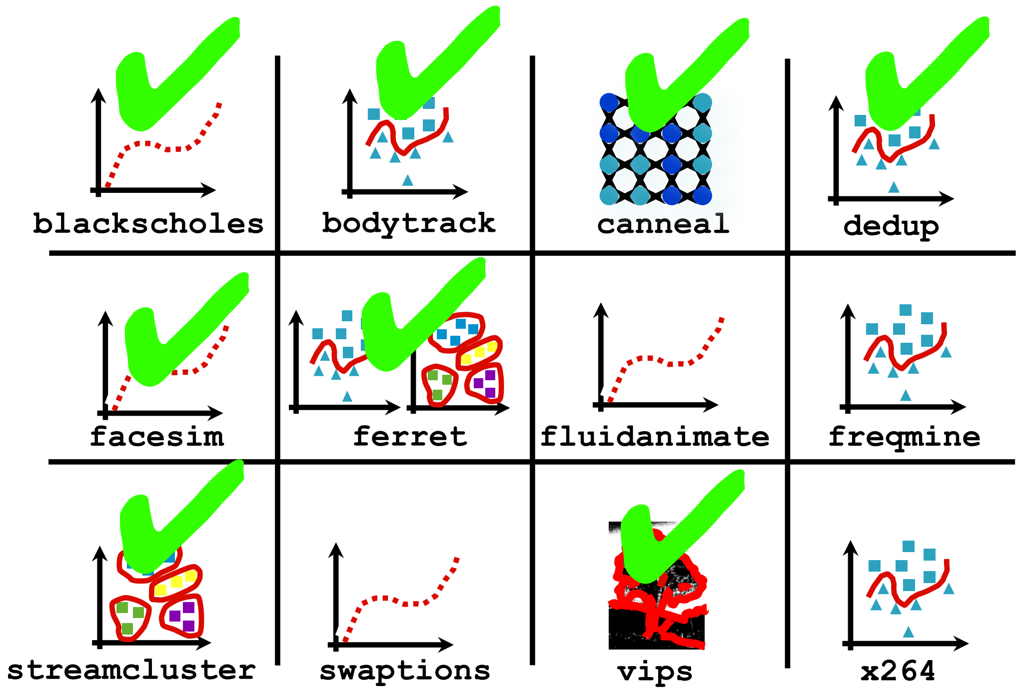
NNet Accelerators

- Flexibility/energy efficiency/robustness/performance

NNets… Again?! [slide: O. Temam]



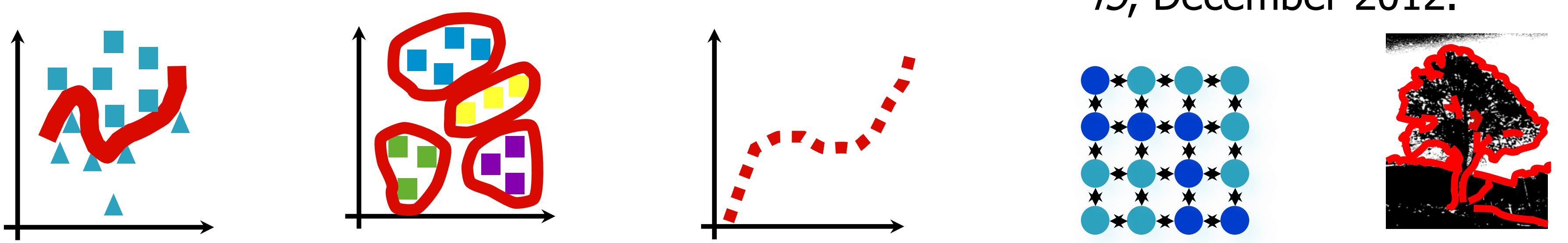
PARSEC Benchmarks



PARSEC

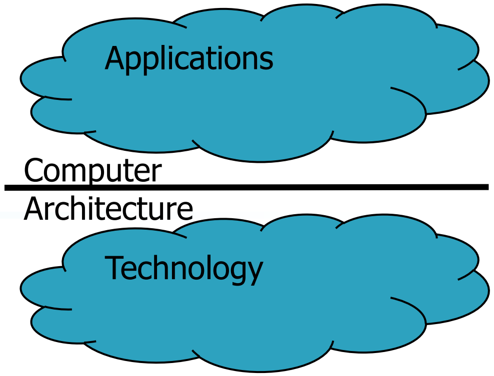
BenchNN: On the Broad Potential Application Scope of Hardware Neural Network Accelerators. T. Chen et al. In *Proc. of the 2012 IISWC 2012*, Nov 2012.

Also: Neural Acceleration for General-Purpose Approximate Programs, H. Esmaeilzadeh et al., *Proceedings of MICRO-45*, December 2012.



Classification **Clustering** **Approximation** **Optimization** **Filtering**

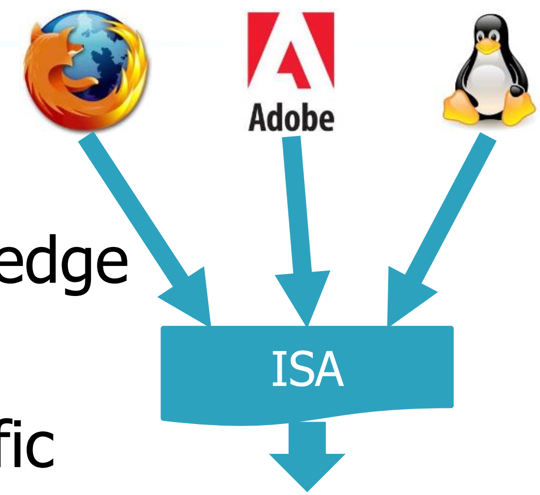
Talk Outline



- Introduction & Motivation
- Neuromorphic applications [IISWC'12]
- Semantic Gap in Neuromorphic Systems
  - Neuromorphic ISA proposal [ASPLOS'11]
  - Digital LIF Spiking Neurons [HPCA'13]
- Conclusions & Future Work

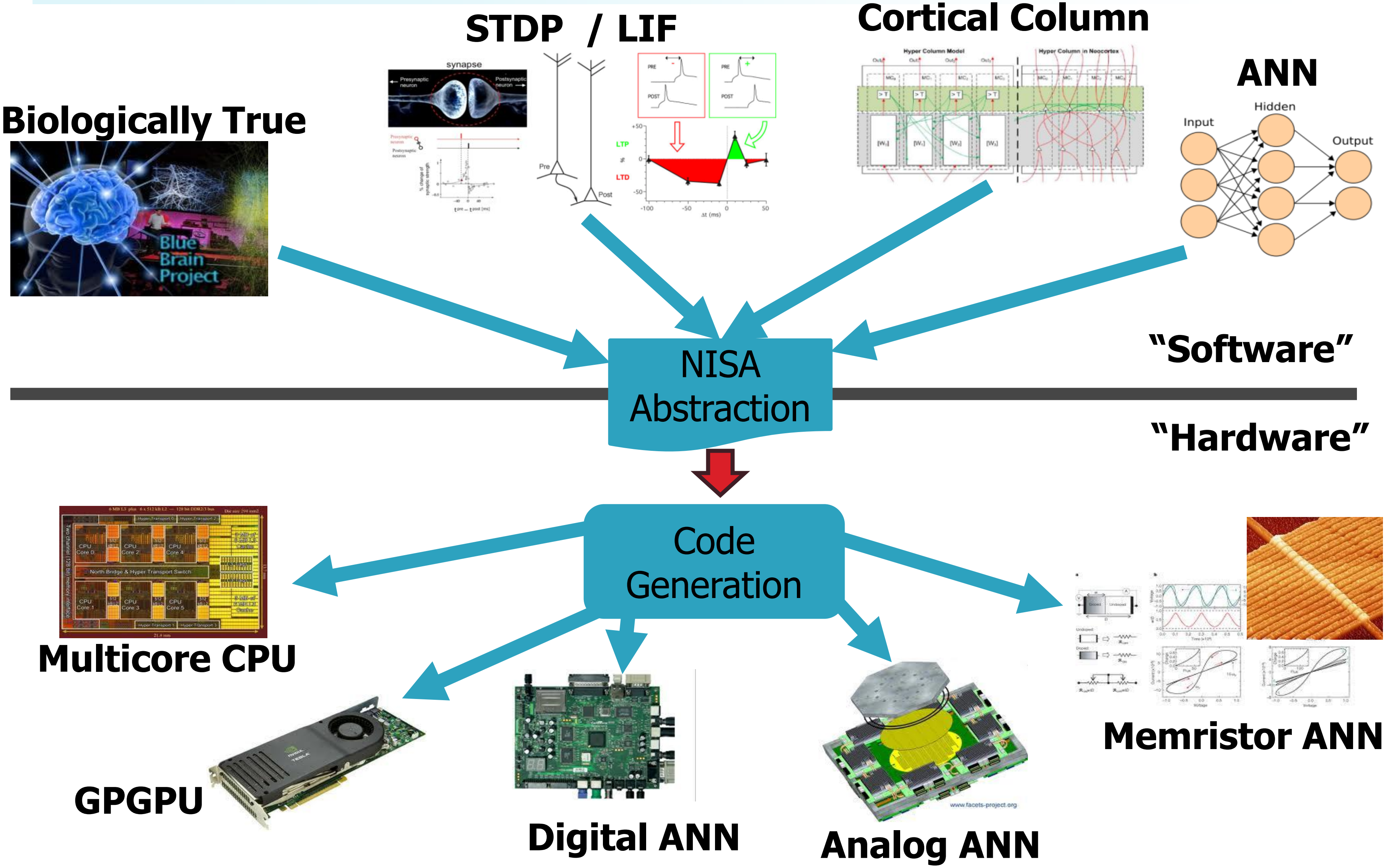
A History Lesson

- Before Instruction Set Architecture…
  - Software depended on hardware knowledge
  - No portability
  - Optimizations were SW / HW pair specific
  - New computer => all new software



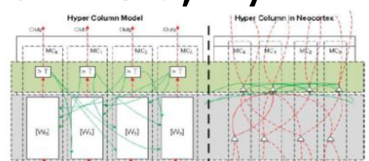
- Gene Amdahl introduces the ISA
  - Contract between SW / HW
  - IBM S/360 line from 1964 to present
  - Independently develop SW and HW
  - Safely optimize, transform SW

NISA proposal [ASPLOS'11]



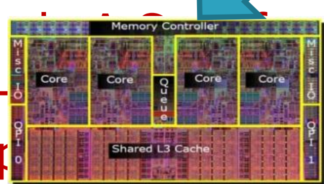
Neuromorphic HW/SW Interface

- Neuromorphic Instruction Set Architecture (NISA)
  - Represents structure and state
  - Automatic deployment/code generation
  - Goals similar to HP Labs COG, PyNN
- Online profiling tools
  - Monitor cortical network and optimize/restructure
- Offline optimizations tools
  - Improve the networks for efficiency and robustness

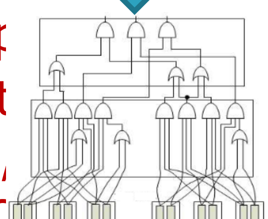


Hypercolumn Model

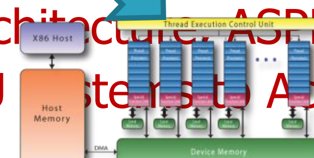
Hardware-Software Interface



CPU



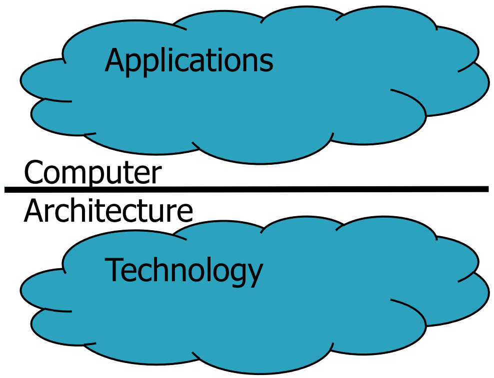
NETLIST



NVIDIA GPU

1. Hashmi et … Neuromorp… on Set Archite… ASPLOS, 2011
2. Nere and H… rofiling Het… Multi-GPU …te…s t… A…celerate Cortically Insp… Algorithms, … 1
3. Nere and Hashmi et al., Simulating C… …orks on Heterogeneous Multi-GPU Systems, JPDC, 2012

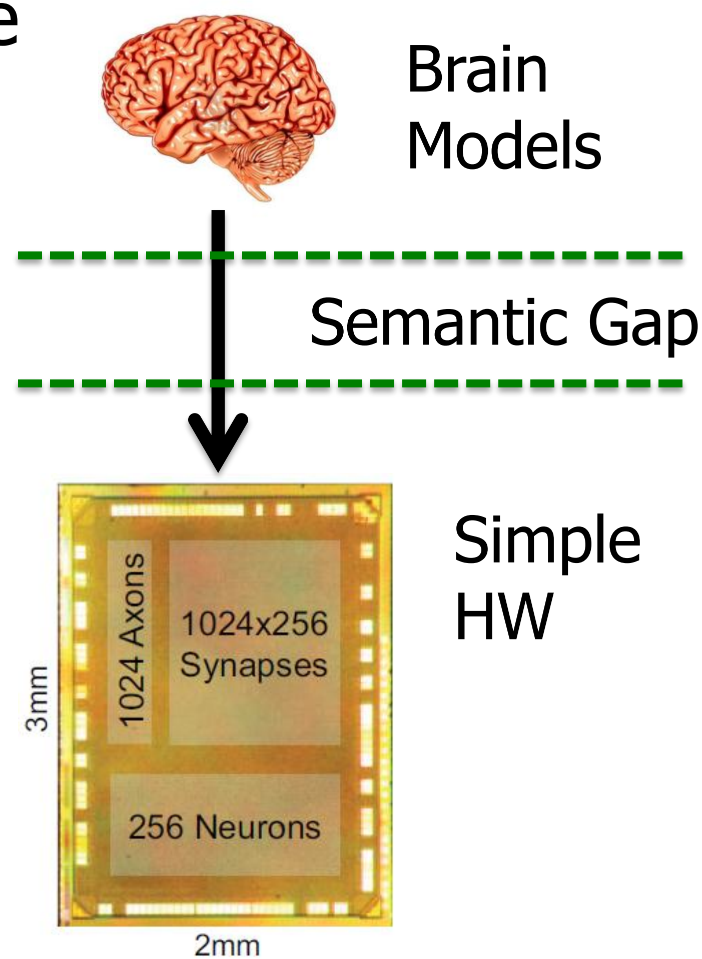
Talk Outline



- Introduction & Motivation
- Neuromorphic applications [IISWC'12]
- Semantic Gap in Neuromorphic Systems
  - Neuromorphic ISA proposal [ASPLOS'11]
  - Digital LIF Spiking Neurons [HPCA'13]
- Conclusions & Future Work

IBM's Neurosynaptic Core

- Digital spiking Neurosynaptic Core Neurons (NCNs)
  - LP CMOS, standard digital logic
  - 256 neurons/core on 4.2mm^2
- "Biologically competitive" energy
  - Few parameters/neuron
  - Binary synapses
  - Linear, no transcendental functions
  - 1kHz operating frequency of NCNs
  - 45pJ/spike



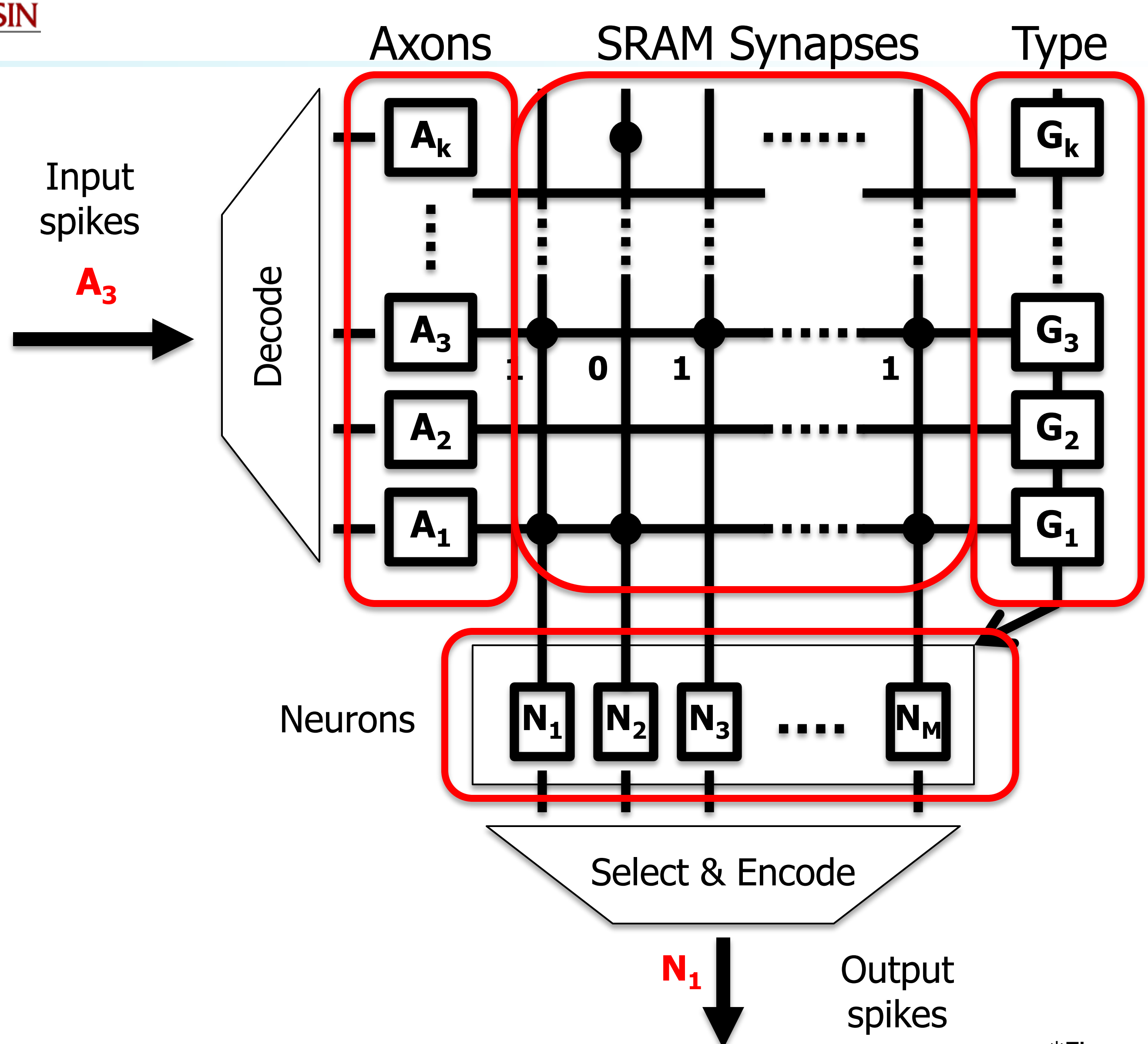
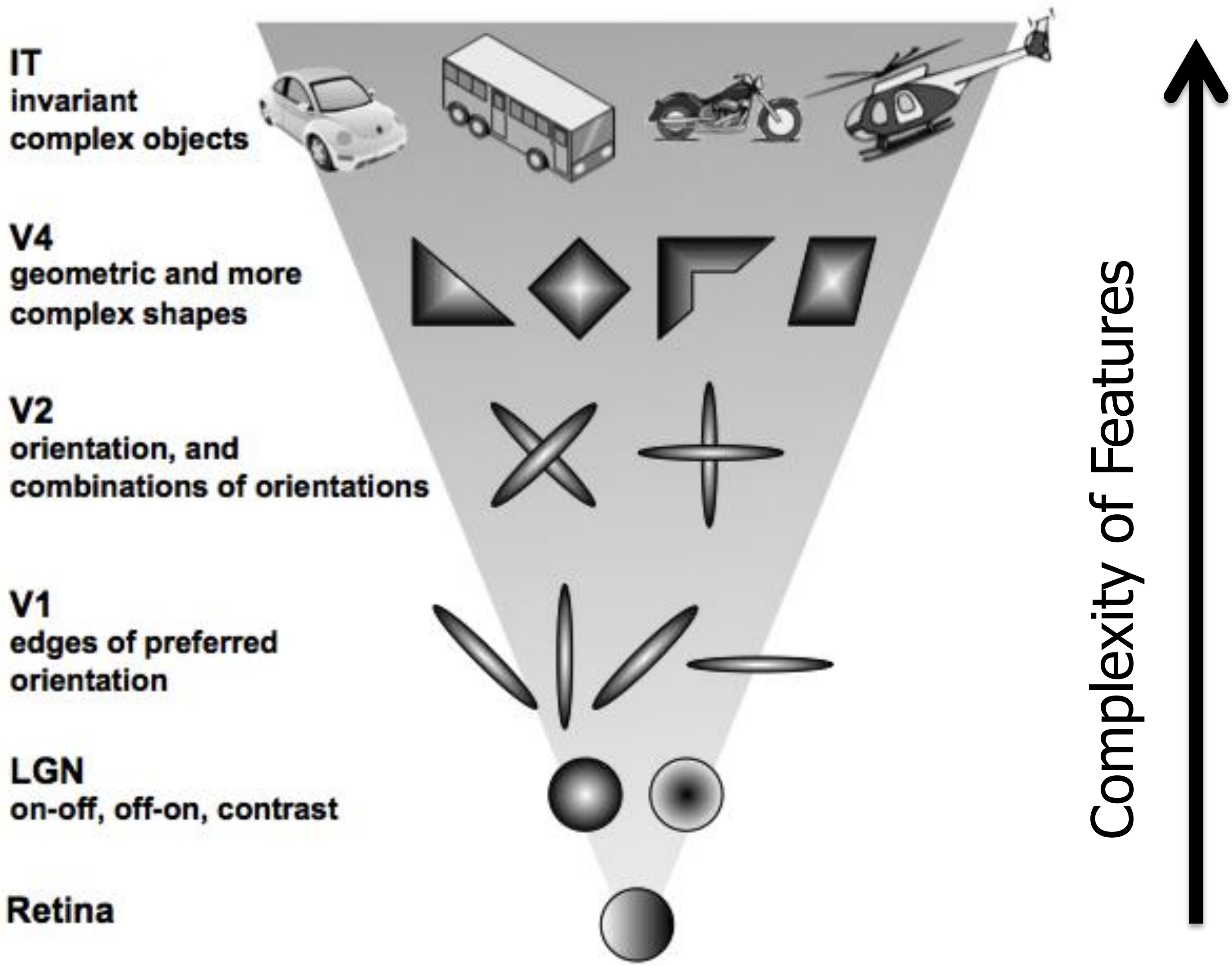
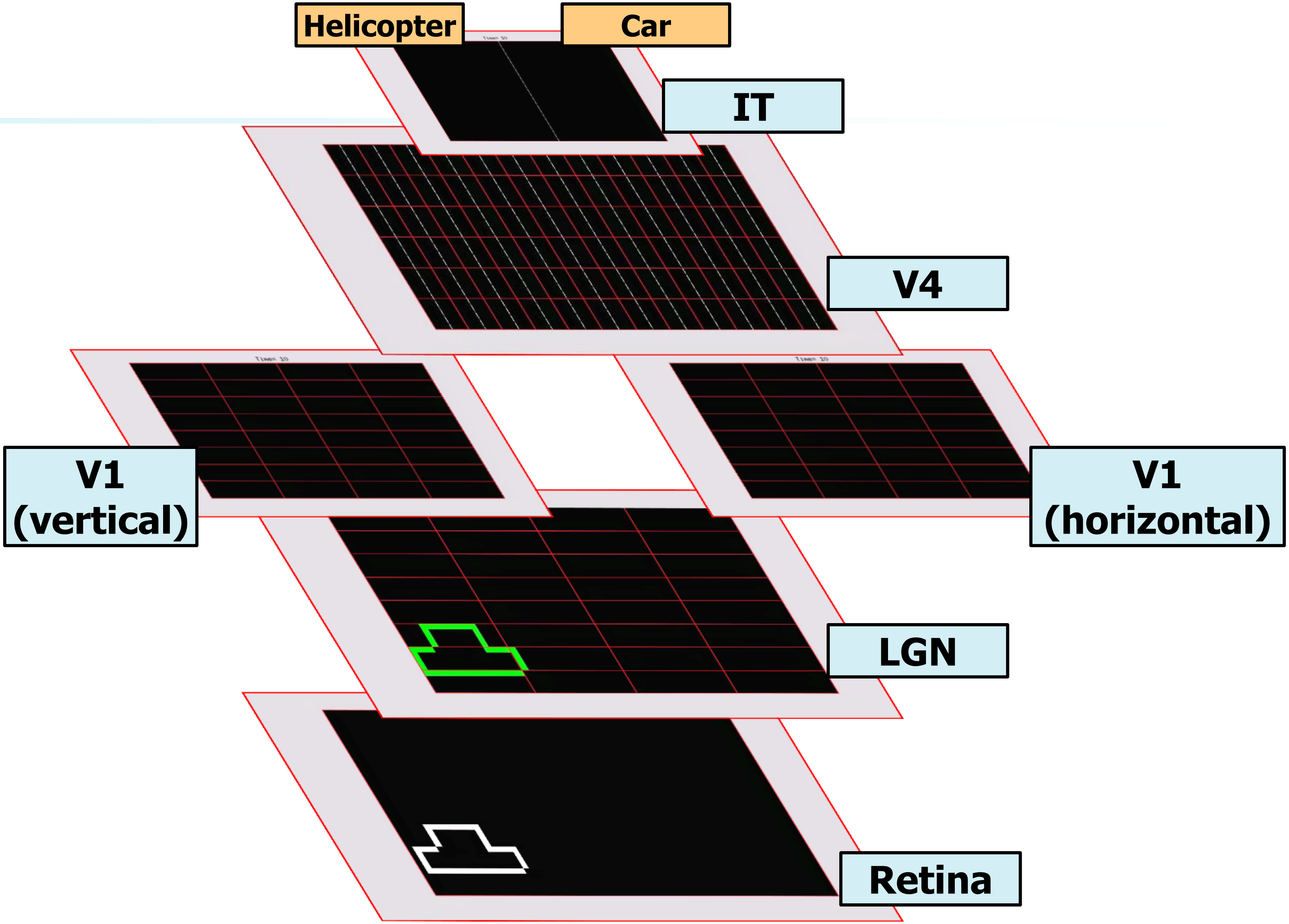


*Figure adapted from Merolla et al.

Visual Cortex

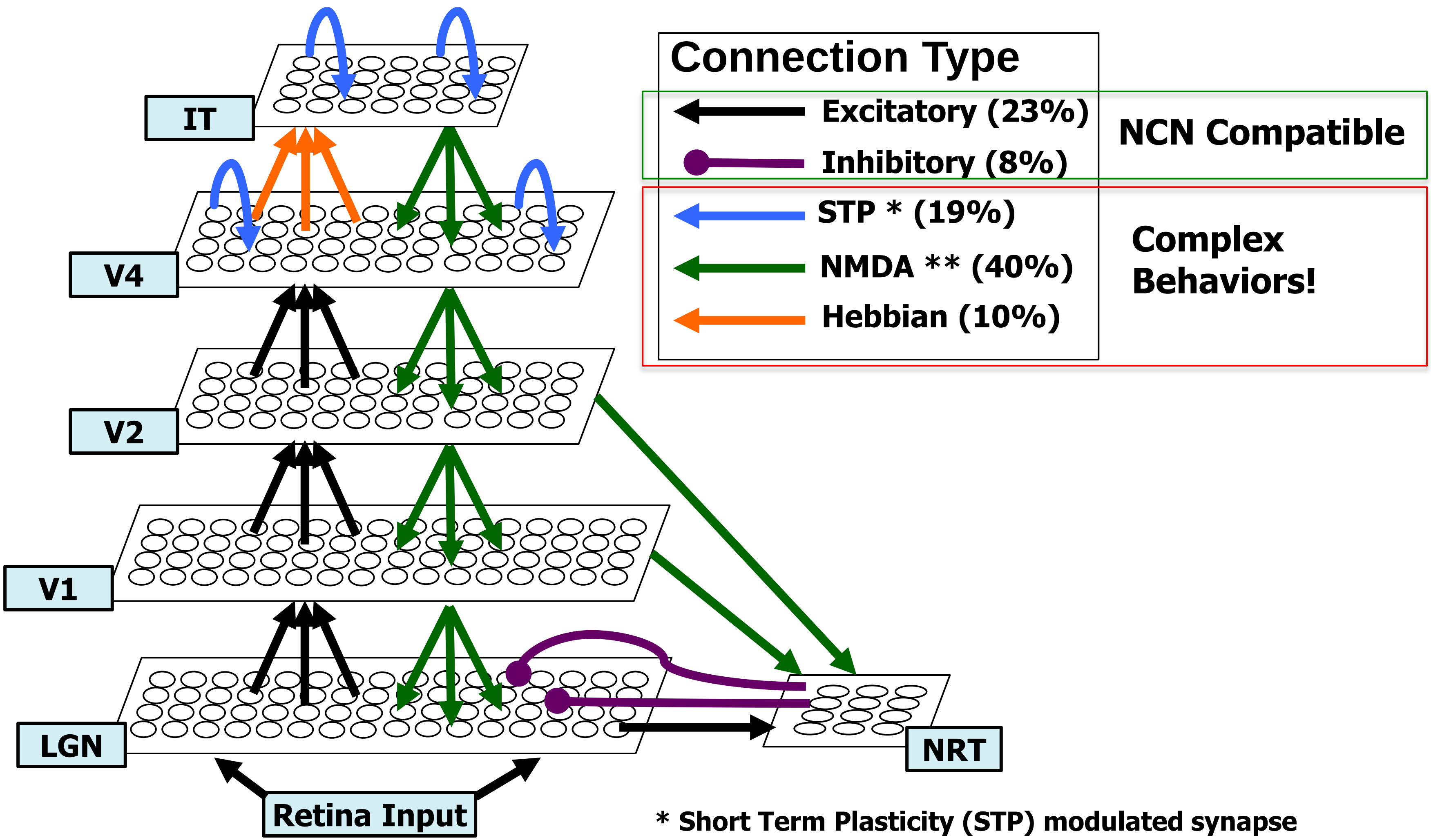



Helicopter
Car
IT
V4
V1 (vertical)
V1 (horizontal)
LGN
Retina

Visual System NNet (VSNN)

- 100,000 modeled neurons
- Applications
  - Invariant object recognition
  - Pattern completion
  - Motion detection/tracking/prediction
  - Noise filtering
- Requires complex neuronal behaviors
  - Not implemented in NCN primitives!

VSNN Architecture

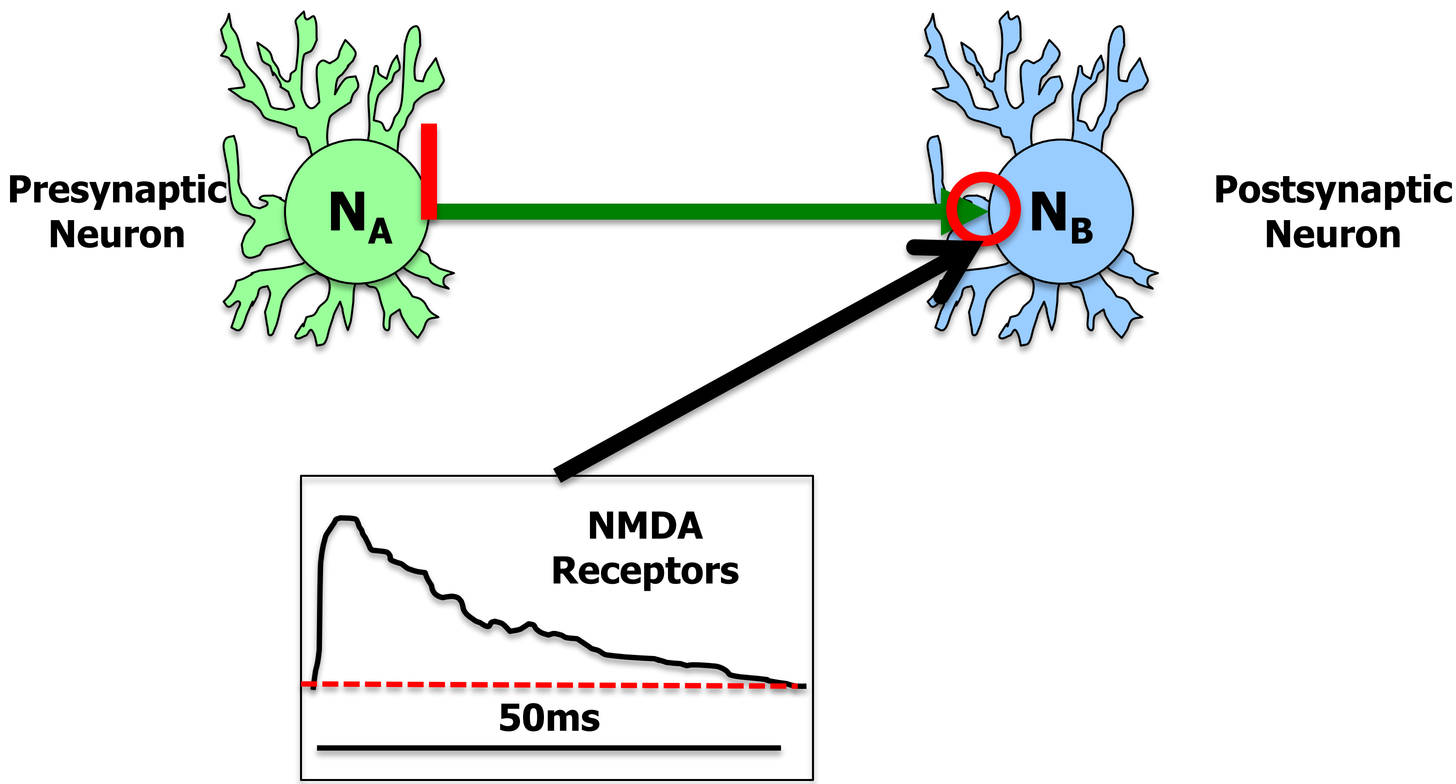


* Short Term Plasticity (STP) modulated synapse

** N-methyl D-aspartate (NMDA) modulated synapses

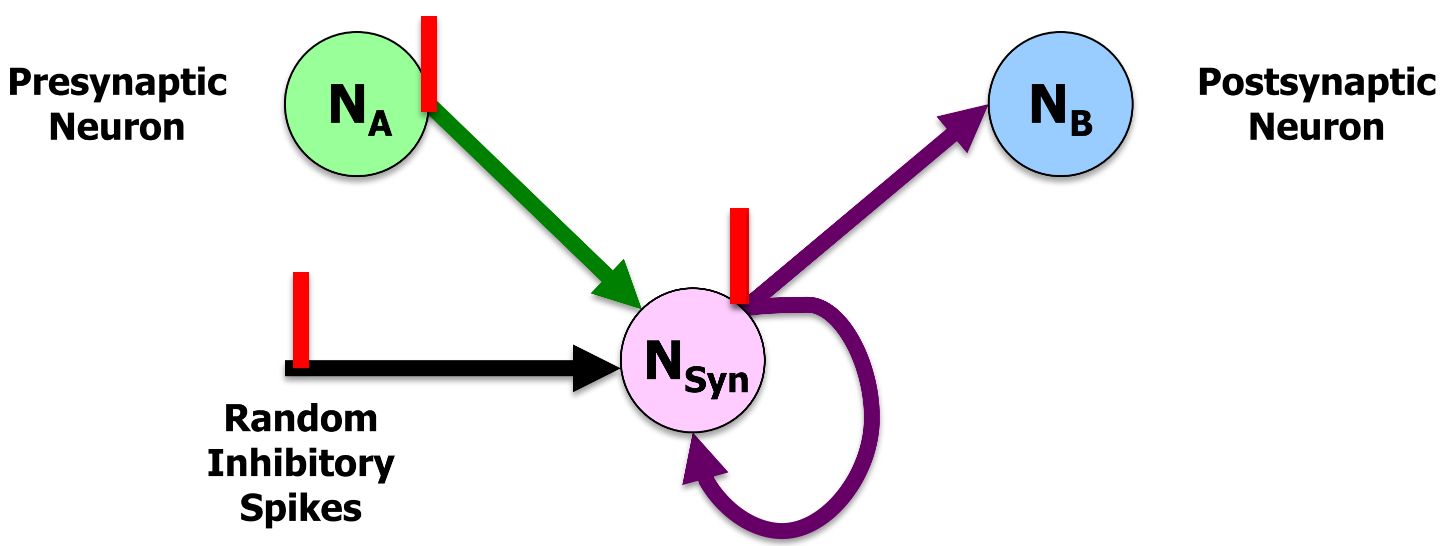
Neuromorphic Semantic Gap

- NCN neurons are very simple (for efficiency)
- Biology incorporates numerous complex behaviors
  - NMDA receptor effects last much longer than 1ms

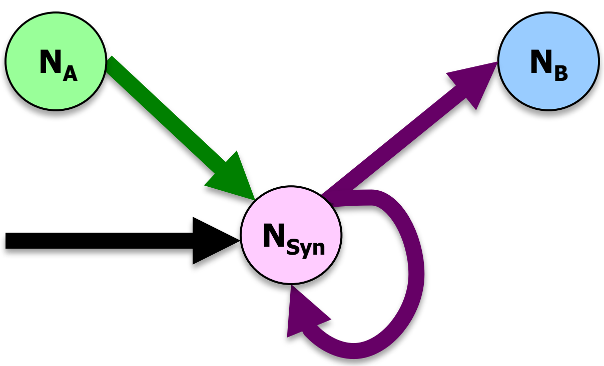


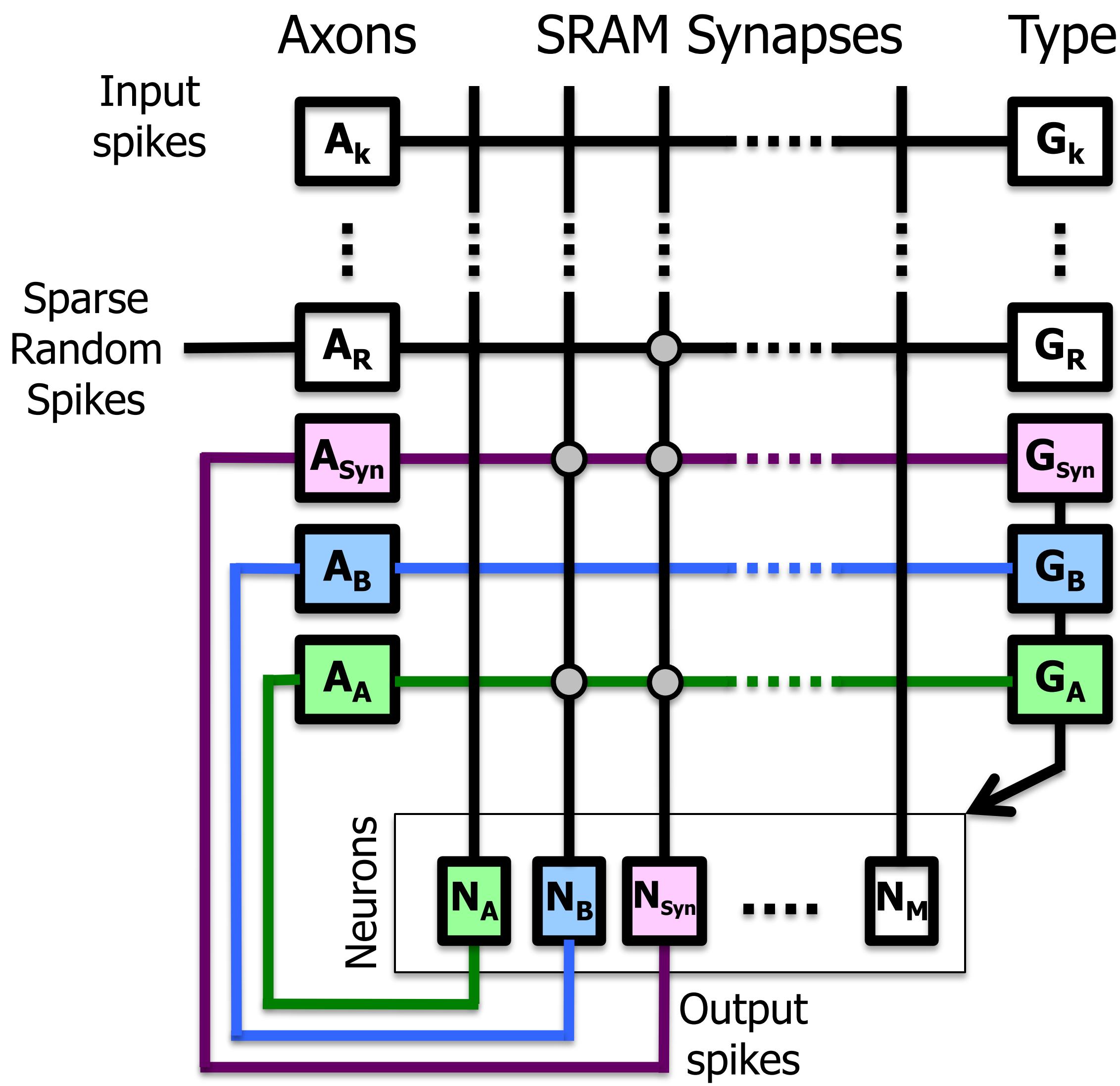
NCN Assembly - NMDA

- Composable circuit of NCN emulates effect

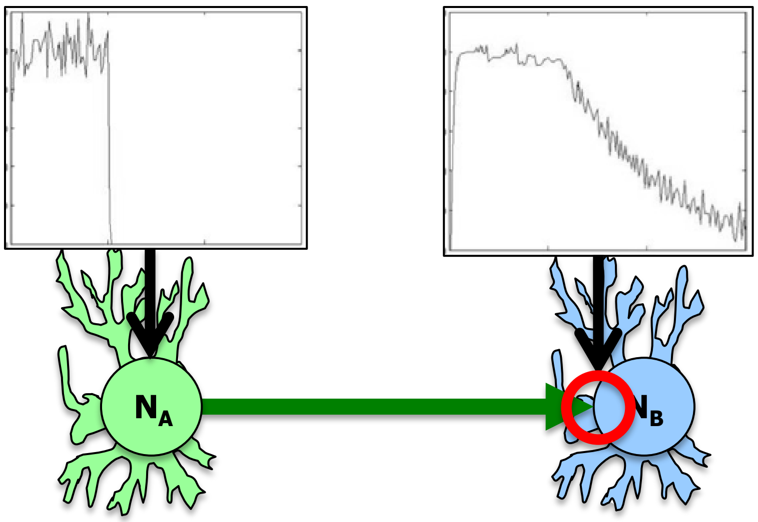


Mapping to IBM NCNs

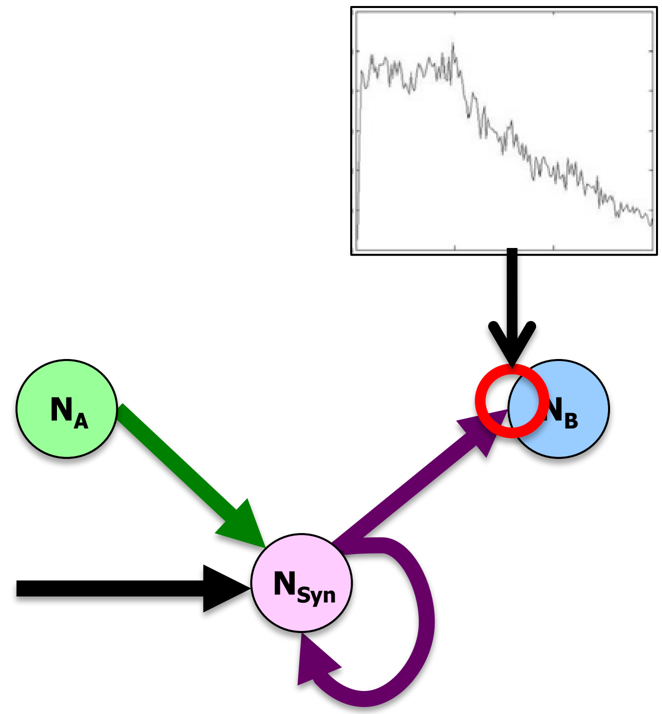




NCN Assembly - NMDA



Complex Neuron/Synapse Model (software)

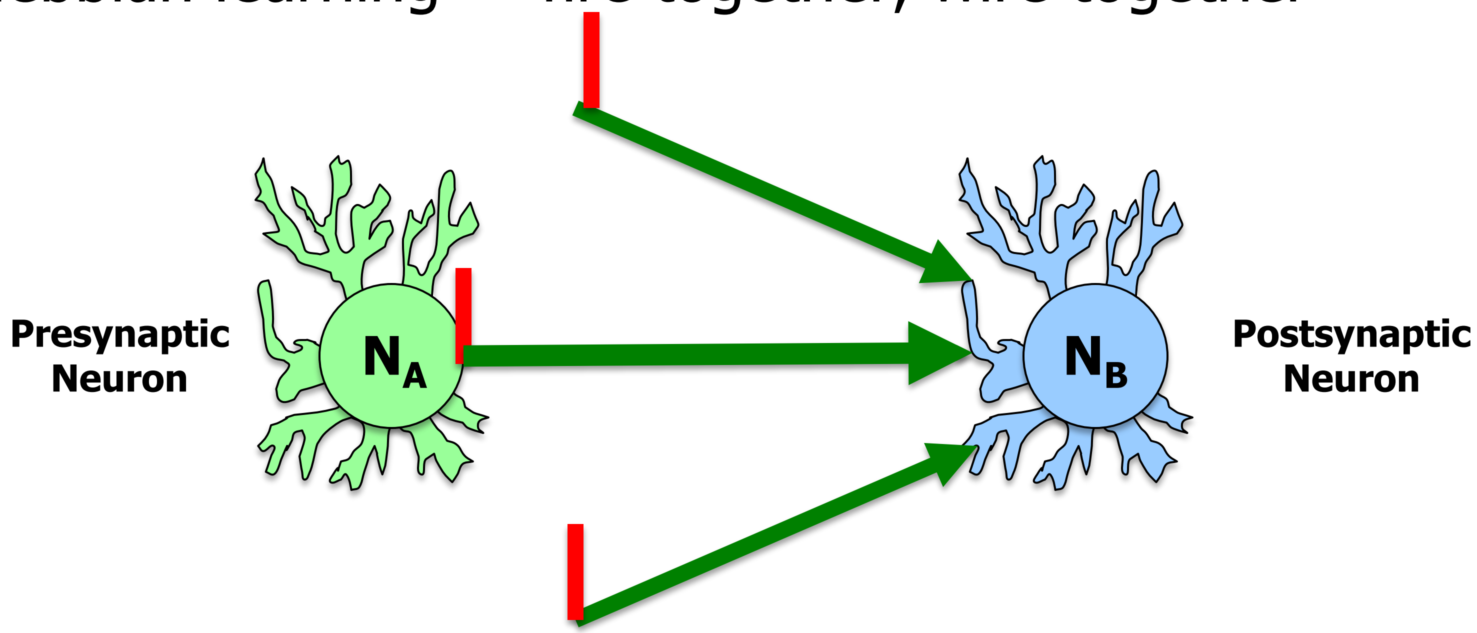


NCN Assembly (Neurosynaptic Core hardware)

- 1 extra NCN/presynaptic neuron area overhead
- ~50*45pJ power overhead (extra spikes)

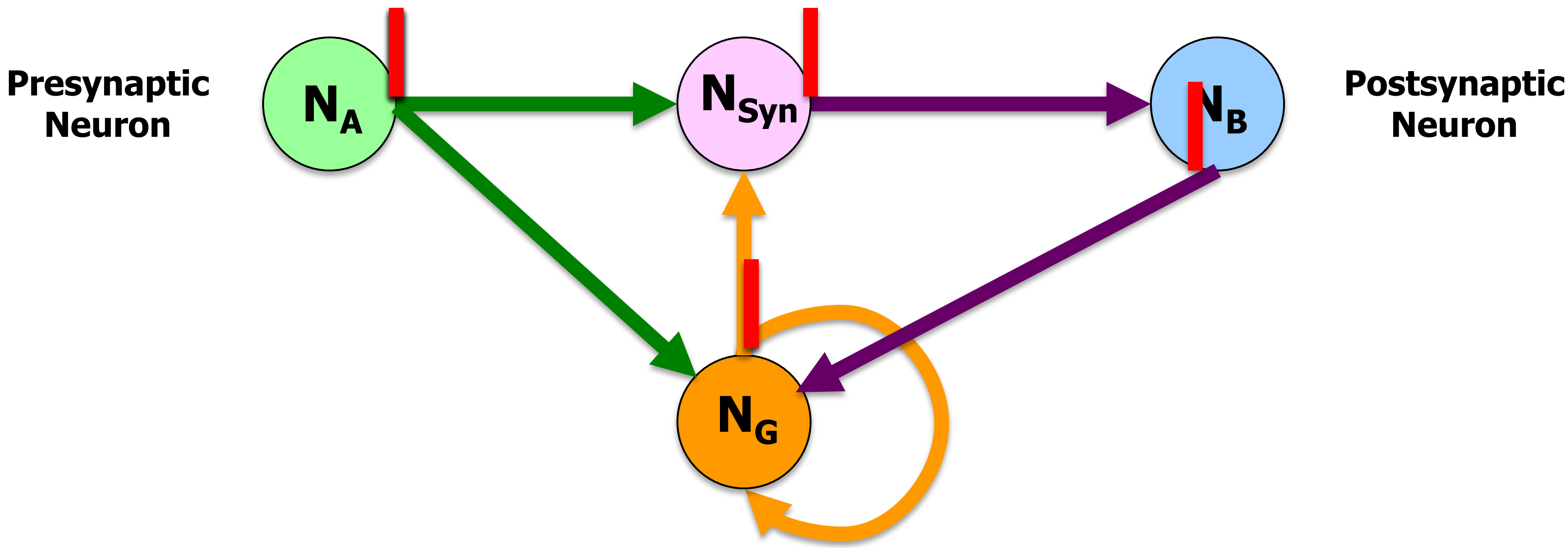
Semantic Gap – Plasticity

- IBM NCN does not support synaptic plasticity*
- Hebbian learning – “fire together, wire together”



*Seo et al. design features two simple online learning rules

Hebbian Learning Assembly

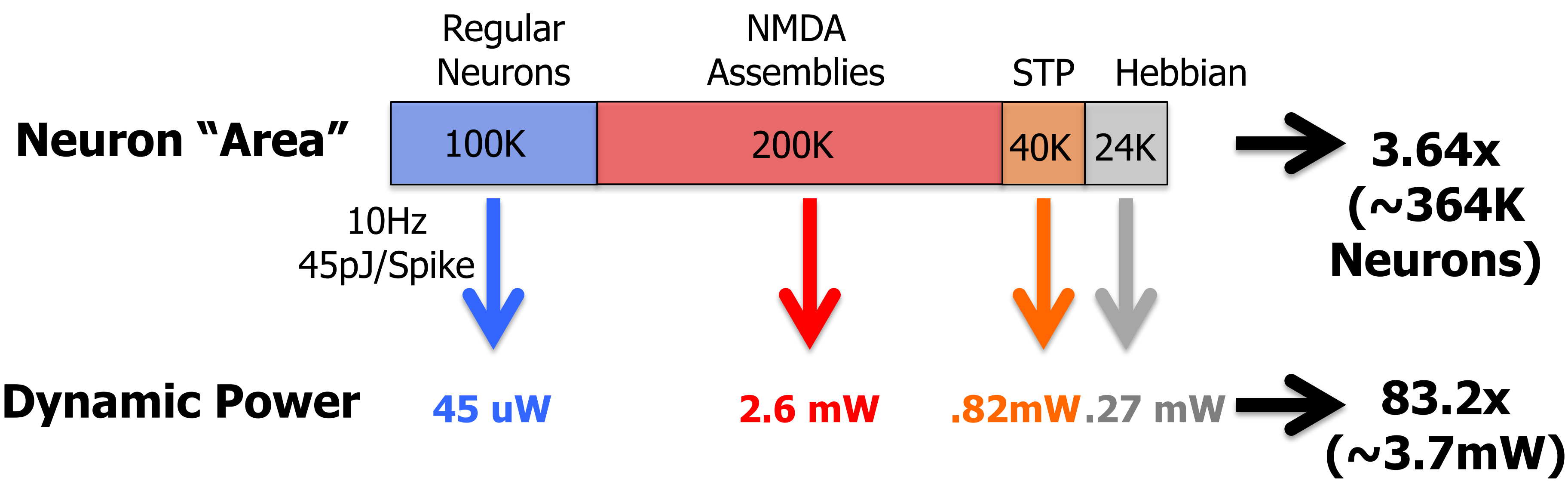


- 2 extra NCNs/synapse
- ~1000*45pJ power overhead/learned synapse

VSNN on Neurosynaptic Core

- "Compiler" replaces complex neurons/synapses with NCN assemblies
  - Deployable on Neurosynaptic Core hardware

- VSNN System Overheads



Conclusions

- Many compelling applications map to neural nets [IISWC'12]
  - Also: *Neural Acceleration for General-Purpose Approximate Programs, H. Esmaeilzadeh et al., Proceedings of MICRO-45, December 2012.*
- Semantic gap between "software" and "hardware"
  - Biological neural networks – complex nonlinear behavior
- Hardware substrates:
  - CPU, GPU, FPGA: compile & optimize [ASPLOS'11]
  - IBM Neurosynaptic Core: map to composable neuronal assemblies
    - Details in [Nere et al. HPCA '13]

Open Questions

- Applications
  - RMS, Approximate computing, robotics/control, …
- Finding the right abstractions/interfaces
  - HP COG? NISA? Multiple NISAs?
  - Theoretical foundations would be helpful
- Building a software ecosystem
  - Compilers, runtimes, libraries, optimizers (static vs. runtime)
- Finding the right hardware primitives
  - Digital LLIF? Analog? Memristor? Parameters, attributes, behavior
  - Online learning, HW vs. SW

Questions?

http://pharm.ece.wisc.edu